DOCUMENT RESUME

ED 067 828 EM 010 179

AUTHOR Goldberg, Adele; Suppes, Patrick
TITLE A Computer-Assisted Instruction Program for Exercises on Finding Axioms. Technical Report Number 186.
INSTITUTION Stanford Univ., Calif. Inst. for Mathematical Studies in Social Science.
SPONS AGENCY National Science Foundation, Washington, D.C.
PUB DATE 23 Jun 72
NOTE 46p.; Psychology and Education Series

EDRS PRICE MF-$0.65 HC-$3.29
DESCRIPTORS Artificial Intelligence; *Computer Assisted Instruction; Computer Programs; Deductive Methods; Instructional Innovation; *Logic; Mathematical Logic; *Program Descriptions

ABSTRACT

An interactive computer-assisted system for teaching elementary logic is described, which was designed to handle formalizations of first-order theories suitable for presentation in a computer-assisted instruction environment. The system provides tools with which the user can develop and then study a nonlogical axiomatic theory along whatever lines he specifies. These tools include a proof-checking program that allows the user to construct derivations by taking advantage of the theorem-proving capabilities of the computer. Results of preliminary investigations using this computer-assisted teaching system in a manner designed to give the student greater control over the organization of his curriculum are summarized, and initial studies on the uses of mechanical theorem provers in teaching about proof construction are outlined. (Author/RH)

ED 067828

NSF

A COMPUTER-ASSISTED INSTRUCTION PROGRAM FOR EXERCISES ON FINDING AXIOMS

SCOPE OF INTEREST NOTICE
The ERIC Facility has assigned this document for processing to: EM SE
In our judgement, this document is also of interest to the clearinghouses noted to the right. Indexing should reflect their special points of view.

BY

ADELE GOLDBERG and PATRICK SUPPES

TECHNICAL REPORT NO. 186

JUNE 23, 1972

PSYCHOLOGY & EDUCATION SERIES

INSTITUTE FOR MATHEMATICAL STUDIES IN THE SOCIAL SCIENCES
STANFORD UNIVERSITY
STANFORD, CALIFORNIA

EM 010179

FILMED FROM BEST AVAILABLE COPY

TECHNICAL REPORTS

PSYCHOLOGY SERIES

INSTITUTE FOR MATHEMATICAL STUDIES IN THE SOCIAL SCIENCES

(Place of publication shown in parentheses; if published title is different from title of Technical Report, this is also shown in parentheses.)

(For reports no. 1 - 44, see Technical Report no. 125.)

50 R. C. Atkinson and R. C. Calfee. Mathematical learning theory. January 2, 1963. (In B. B. Wolman (Ed.), Scientific Psychology. New York: Basic Books, Inc., 1965. Pp. 254-275)

51 P. Suppes, E. Crothers, and R. Weir. Application of mathematical learning theory and linguistic analysis to vowel phoneme matching in Russian words. December 28, 1962.

52 R. C. Atkinson, R. Calfee, G. Sommer, W. Jeffrey and R. Shoemaker. A test of three models for stimulus compounding with children. January 29, 1963. (J. exp. Psychol., 1964, 67, 52-58)

53 E. Crothers. General Markov models for learning with inter-trial forgetting. April 8, 1963.

54 J. L. Myers and R. C. Atkinson. Choice behavior and reward structure. May 24, 1963. (Journal math. Psychol., 1964, 1, 170-203)

55 R. E. Robinson. A set-theoretical approach to empirical meaningfulness of measurement statements. June 10, 1963.

56 E. Crothers, R. Weir and P. Palmer. The role of transcription in the learning of the orthographic representations of Russian sounds. June 17,. 1963.

57 P. Suppes. Problems of optimization in learning a list of simple items. July 22, 1963. (In Maynard W. Shelly, II and Glenn L. Bryan (Eds.), Human Judgments and Optimality. New York: Wiley. 1964. Pp. 116-126)

58 R. C. Atkinson and E. J. Crothers. Theoretical note: all-or-none learning and intertrial forgetting. July 24, 1963.

59 R. C. Calfee. Long-term behavior of rats under probabilistic reinforcement schedules. October 1, 1963.

60 R. C. Atkinson and E. J. Crothers. Tests of acquisition and retention, axioms for paired-associate learning. October 25, 1963. (A comparison of paired-associate learning models having different acquisition and retention axioms, J. math. Psychol., 1964, 1, 285-315)

61 W. J. McGill and J. Gibbon. The general-gamma distribution and reaction times. November 20, 1963. (J. math. Psychol., 1965, 2, 1-18)

62 M. F. Norman. Incremental learning on random trials. December 9, 1963. (J. math. Psychol., 1964, 1, 336-351)

63 P. Suppes. The development of mathematical concepts in children. February 25, 1964. (On the behavioral foundations of mathematical concepts. Monographs of the Society for Research in Child Development, 1965, 30, 60-96)

64 P. Suppes. Mathematical concept formation in children. April 10, 1964. (Amer. Psychologist, 1966, 21, 139-150)

65 R. C. Calfee, R. C. Atkinson, and T. Shelton, Jr. Mathematical models for verbal learning. August 21, 1964. (In N. Wiener and J. P. Schoda (Eds.), Cybernetics of the Nervous System: Progress in Brain Research. Amsterdam, The Netherlands: Elsevier Publishing Co., 1965. Pp. 333-349)

66 L. Keller, M. Cole, C. J. Burke, and W. K. Estes. Paired associate learning with differential rewards. August 20, 1964. (Reward and information values of trial outcomes in paired associate learning. (Psychol. Monogr., 1965, 79, 1-21)

67 M. F. Norman. A probabilistic model for free-responding. December 14, 1964.

68 W. K. Estes and H. A. Taylor. Visual detection in relation to display size and redundancy of critical elements. January 25, 1965, Revised 7-1-65. (Perception and Psychophysics, 1966, 1, 9-16)

69 P. Suppes and J. Donio. Foundations of stimulus-sampling theory for continuous-time processes. February 9, 1965. (J. math. Psychol., 1967, 4, 202-225)

70 R. C. Atkinson and R. A. Kinchla. A learning model for forced-choice detection experiments. February 10, 1965. (Br. J. math stat. Psychol., 1965, 18, 184-206)

71 E. J. Crothers. Presentation orders for items from different categories. March 10, 1965.

72 P. Suppes, G. Green, and M. Schlag-Rey. Some models for response latency in paired-associates learning. May 5, 1965. (J. math. Psychol., 1966, 3, 99-128)

73 M. V. Levine. The generalization function in the probability learning experiment. June 3, 1965.

74 D. Hansen and T. S. Rodgers. An exploration of psycholinguistic units in initial reading. July 6, 1965.

75 B. C. Arnold. A correlated urn-scheme for a continuum of responses. July 20, 1965.

76 C. Izawa and W. K. Estes. Reinforcement-test sequences in paired-associate learning. August 1, 1965. (Psychol. Reports, 1966, 18, 879-919)

77 S. L. Biehart. Pattern discrimination learning with Rhesus monkeys. September 1, 1965. (Psychol. Reports, 1966, 19, 311-324)

78 J. L. Phillips and R. C. Atkinson. The effects of display size on short-term memory. August 31, 1965.

79 R. C. Atkinson and R. M. Shiffrin. Mathematical models for memory and learning. September 20, 1965.

80 P. Suppes. The psychological foundations of mathematics. October 25, 1965. (Colloques Internationaux du Centre National de la Recherche Scientifique. Editions du Centre National de la Recherche Scientifique. Paris: 1967. Pp. 213-242)

81 P. Suppes. Computer-assisted instruction in the schools: potentialities, problems, prospects. October 29, 1965.

82 R. A. Kinchla, J. Townsend, J. Yellott, Jr., and R. C. Atkinson. Influence of correlated visual cues on auditory signal detection. November 2, 1965. (Perception and Psychophysics, 1966, 1, 67-73)

83 P. Suppes, M. Jerman, and G. Groen. Arithmetic drills and review on a computer-based teletype. November 5, 1965. (Arithmetic Teacher, April 1966, 303-309.

84 P. Suppes and L. Hyman. Concept learning with non-verbal geometrical stimuli. November 15, 1965.

85 P. Holland. A variation on the minimum chi-square test. (J. math. Psychol., 1967, 3, 377-413).

86 P. Suppes. Accelerated program in elementary-school mathematics -- the second year. November 22, 1965. (Psychology in the Schools, 1966, 3, 294-307)

87 P. Lorenzen and F. Binford. Logic as a dialogical game. November 29, 1965.

88 L. Keller, W. J. Thomson, J. R. Tweedy, and R. C. Atkinson. The effects of reinforcement interval on the acquisition of paired-associate responses. December 10, 1965. (J. exp. Psychol., 1967, 73, 268-277)

89 J. I. Yellott, Jr. Some effects on noncontingent success in human probability learning. December 15, 1965.

90 P. Suppes and G. Groen. Some counting models for first-grade performance data on simple addition facts. January 14, 1966. (In J. M. Scandura (Ed.), Research in Mathematics Education. Washington, D. C.: NCTM, 1967. Pp. 35-43.

91 P. Suppes. Information processing and choice behavior. January 31, 1966.

92 G. Groen and R. C. Atkinson. Models for optimizing the learning process. February 11, 1966. (Psychol. Bulletin, 1966, 66, 309-320)

93 R. C. Atkinson and D. Hansen. Computer-assisted instruction in initial reading: Stanford project. March 17, 1966. (Reading Research Quarterly, 1966, 2, 5-25)

94 P. Suppes. Probabilistic inference and the concept of total evidence. March 23, 1966. (In J. Hintikka and P. Suppes (Eds.), Aspects of Inductive Logic. Amsterdam: North-Holland Publishing Co., 1966. Pp. 49-65.

95 P. Suppes. The axiomatic method in high-school mathematics. April 12, 1966. (The Role of Axiomatics and Problem Solving in Mathematics. The Conference Board of the Mathematical Sciences, Washington, D. C. Ginn and Co., 1966. Pp. 69-76.

(Continued on inside back cover)

ED 067828

A COMPUTER-ASSISTED INSTRUCTION PROGRAM FOR EXERCISES ON FINDING AXIOMS

by

Adele Goldberg and Patrick Suppes

TECHNICAL REPORT NO. 186

June 23, 1972

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
OFFICE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL OFFICE OF EDUCATION POSITION OR POLICY

PSYCHOLOGY AND EDUCATION SERIES

Reproduction in Whole or in Part Is Permitted for

Any Purpose of the United States Government

INSTITUTE FOR MATHEMATICAL STUDIES IN THE SOCIAL SCIENCES

STANFORD UNIVERSITY

STANFORD, CALIFORNIA

A Computer-assisted Instruction Program for Exercises on Finding Axioms*

by Adele Goldberg and Patrick Suppes

This paper describes an interactive computer-assisted system for teaching elementary mathematical logic, which was designed to handle formalizations of first-order theories suitable for presentation in a computer-assisted instruction (CAI) environment. The system provides tools with which the user can develop and then study a nonlogical axiomatic theory along whatever lines he specifies. These tools include a proof-checking program that allows the user to construct derivations by taking advantage of the theorem-proving capabilities of the computer.

Results of preliminary investigations using this computer-assisted teaching system in a manner designed to give the student greater control over the organization of his curriculum are summarized in Section 2, and Section 3 outlines initial studies on the uses of mechanical theorem provers in teaching about proof construction.

---

* This research has been supported by the National Science Foundation under Grant NSF GJ-443X2. We very much appreciate the assistance of Stephen A. Weyer in preparing this report for publication.

1. Basic Instructional Capabilities

The instructional system is written in the LISP programming language for a DEC PDP-10 computer at the CAI Laboratory of the Institute for Mathematical Studies in the Social Sciences (IMSSS), Stanford University. Figure 1, a block diagram of the basic system, views the program as an interpreter made up of two components. One, call it C, is a set of inference rules and proof procedures used in constructing derivations (DERIVE mode). It is defined in Language A (AXIOMATIZE mode). With A, the user is able to specify the vocabulary and axioms and to derive new rules of inference from axioms and established theorems. The names of the axioms, theorems and new rules, with instructions on how to use these rules, are learned by A and added to C, which is further augmented by the basic logical system. Specifically, this includes primitive rules to support a formalization of first-order predicate calculus with identity--quantifier rules, the rule of detachment (called "affirm the antecedent" and abbreviated AA), rules governing the identity relation, proper substitution (PS) and replacement, and procedures to construct conditional and indirect proofs (CP and IP).

Insert Figure 1 about here

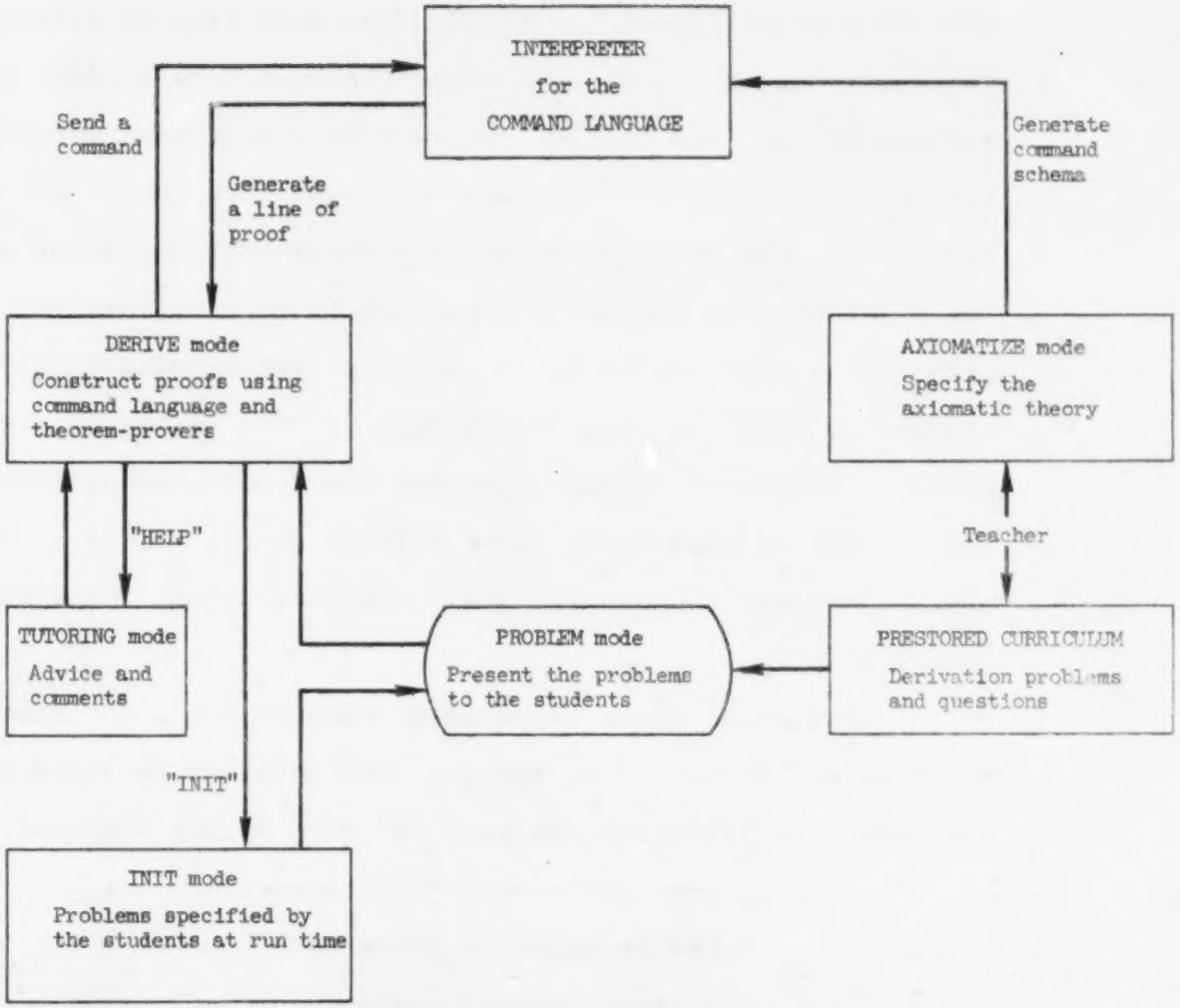

Fig. 1. Block diagram of the instructional system.

In the DERIVE mode, the interpreter takes on the role of a proof checker to verify that each step of a proof constructed by the user is a valid consequent of a set of statements of the theory. In such an interactive system, the user indicates a proof step by typing the name of a rule in C. The rule may refer to several previous lines of the proof and may include a reference to an occurrence of a term in a previous line. A properly formulated rule is a request to have the program generate a new line of the proof. The proof checker examines the format and intended application of each rule. If a rule is poorly formed, the user receives a message that explains what error was committed.

Figure 2 shows a simple example of a proof constructed within the system. Any information typed by the user is underlined; the rest is typed by the computer.

Insert Figure 2 about here

In the proof in Figure 2, the user typed WP to indicate that he wanted to type a working premise. Derived rule LC used in line (3) is a special case of the classical sentential rule of simplification; we use it to infer the left conjunct of a conjunction. The rule CP, for conditional proof, is a request to generate a conditional

```
PROVE      (P->Q)->((P & R)->Q)
:WP        (1)        P->Q
:WP        (2)             P & R
:2LC       (3)             P
:1,3AA     (4)             Q
:2,4CP     (5)        (P & R)->Q
:1,5CP     (6)   (P->Q)->((P & R)->Q)
CORRECT...
```

Fig. 2. Sample proof (underlining indicates student input).

statement from a working premise and some line already in the proof. The production of a line matching the problem expression is not sufficient to guarantee that the user has constructed a complete proof. Any working premises that have been introduced must be discharged in proper order by a conditional or indirect proof.

The proof checker can provide error messages for the rules it learns from the user. The same analysis routines that process rules and generate new lines of the proof are also used to compute error messages. These error-analysis procedures generate explanations of how each rule may be applied. The explanations include sample proof steps which, if possible, reference lines already in the user's proof. In this manner, the user can receive immediate feedback to help him reexamine the structure of the theory he himself has specified.

Sample recursive error messages for sentential logic are shown in Figure 3. The mnemonic DC refers to the principle of inference modus tollens. Hopefully no single student would ever make so many mistakes in a single exercise.

Insert Figure 3 about here

Labeling the user as either a teacher or student is

```
PROVE     (P->Q)->((NOT Q -> NOT P)->NOT P)
:WP         (1)        P->Q
:WP         (2)              NOT Q -> NOT P
:WP         (3)                   P
:3,1AA
        LINE 3 MUST BE A CONDITIONAL
:1,2AA
        LINE 2 MUST BE THE ANTECEDENT OF LINE 1
:1,3AA      (4)                   Q
:2,3DC      (5)                   NOT Q
:2,4,3IP
        LINE 2 WAS NOT THE LAST WORKING PREMISE INTRODUCED
:3,4,2IP
        LINE 4 IS NOT THE DENIAL OF LINE 2
:3,4,5IP    (6)             NOT P
:5,2CP
        LINE 5 IS NOT A WORKING PREMISE
:2,6CP      (7)        (NOT Q -> NOT P)-> NOT P
:1,7CP      (8)  (P->Q)->((NOT Q -> NOT P)->NOT P)
CORRECT...
```

Fig. 3. Sample error messages.

unnecessary. It is possible for the user to devise a curriculum for teaching others about the axiomatic system he has specified, so the user might be a teacher writing a computer-assisted course or a student experimenting with newly acquired skills in proof construction. Although the teacher may prescribe a curriculum, his student may interrupt the flow of exercises to make up his own exercises.

As shown in Figure 4, the student indicates his intention to alter the curriculum by typing the instruction INIT. He may avail himself of this option to test any ideas he has acquired about proof strategies or about the provability of some expression. He may repeat an exercise for practice, try more difficult derivations than those given him or prove lemmas that would help make the teacher-defined exercises easier to solve. He thus can alter the curriculum to suit his individual instructional needs. In Figure 4 we see how a student derived a new rule of inference from a lemma he has proved and then used the rule and two theorems to complete the exercise given to him. The two theorems are shown at the bottom of Figure 4. Error messages computed for the derived rule have also been included.

Insert Figure 4 about here

The student is not required to follow a particular solution path in order to construct a valid derivation. He is always free to explore within the bounds set by the inference rules, to ask questions about rules and to try any approaches he thinks are promising for finding a proof. However, the degree of student control over the curriculum can be changed at the teacher's discretion. By varying the degree of freedom given the student, we will in time be able systematically to test the comparative effectiveness of student-controlled versus fixed-branching CAI schemes.

In the present version of the instructional system, the teacher can block use of the INIT command, especially where consecutive experience with a set of exercises is desirable, or the teacher can indicate in the curriculum that a problem must be solved with or without the use of certain rules. Once the student has completed a valid solution, the proof checker chains through the sequence of rules that entered into the derivation to determine whether the student constructed the proof within restrictions specified by the teacher. If the student has not constructed a solution within the required restrictions, the proof checker will ask him to repeat the exercise.

```
DERIVE      (NOT P)->(P -> Q)
:INIT
          YOU CAN NOW REQUEST AN EXERCISE OR
          DERIVE A NEW RULE OF INFERENCE.
          TYPE  FIN  WHEN YOU ARE DONE.

:PROVE:  (P -> Q)->((Q -> R)->(P -> R))

:WP          (1)        P -> Q

:WP          (2)             Q -> R

:WP          (3)                  P

:1,3AA       (4)              Q

:2,4AA       (5)                  R

:3,5CP       (6)             P -> R

:2,6CP       (7)       (Q -> R)->(P -> R)

:1,7CP       (8)  (P -> Q)->((Q -> R)->(P -> R))

CORRECT...
NAME:LEM

:RULE
NAME:HS
FROM:LEM

:FIN
```

Fig. 4. Use of the INIT instruction.

```
NOW REDO THE EXERCISE YOU INTERRUPTED

DERIVE      (NOT P)->(P -> Q)

PS:TH2      ((NOT Q)->(NOT P))->(P -> Q)
::        (1)  ((NOT Q)->(NOT P))->(P -> Q)

PS:TH1      P->(Q -> P)
::P:NOT P
::Q:NOT Q
::        (2)  (NOT P)->((NOT Q)->(NOT P))

:1,2HS
        HS REQUIRES 3 LINE NUMBERS

:WP       (3)       NOT P

:1,2,3HS
        THE ANTECEDENT OF LINE 2 MUST BE
        THE CONSEQUENT OF LINE 1

:2,1,3HS (4)        P -> Q

:3,4CP    (5)  (NOT P)->(P -> Q)

CORRECT...
```

Fig. 4, continued.

In summary, the instructional system was designed to increase the level of active participation by the student. A student can specify a first-order theory and build his own command language with which to construct proofs. He can then (a) make up his own exercises, (b) communicate with a proof checker to verify whether his solutions are correct, (c) work on an exercise without interference, i.e., he may try any solution path regardless of whether it approaches a successful proof, and (d) receive immediate feedback on errors. Furthermore, he can ask about previously learned material and receive advice and comments on his work which take into account only the material that he knows, i.e., material that he was taught or that he developed. This flexibility was made possible only by developing a system that was not restricted to a fixed curriculum or a fixed language for communicating with the students. Consequently, the system can be adapted to the instructional needs of a large class of users and a wide range of possible curriculums.

2. The Finding-Axioms Exercises

The possible uses for this system are varied. The resources of the program allow the user to axiomatize the theoretical structures of some elementary domain of science. For example, a graduate student in logic is using the program to study axiomatizations of elementary

geometry; another is using it to develop a curriculum for the predicate calculus and elementary number theory.

A program like this is suitable for problem-solving tasks of varying levels of complexity, such as the Finding-Axioms Exercises, which are modeled after the famous R. L. Moore method of instruction. The students were either in an elementary course in logic at Stanford or 12-year-old seventh graders from a junior high school in Palo Alto, California. A description of this task and the results of actual classroom use of the instructional program follow.

A Finding-Axioms Exercise consists of a list of well-formed formulas of a formal theory. The student must select at most N formulas from which the rest can be derived and show that the selection is correct by carrying out the derivations. The student is encouraged to establish a definite order of the formulas to be proved so that one derived formula can be effectively used in deriving another.

For this exercise, the logical rules of inference already discussed were made available to the student. The only difference in format was the additional ability to reference the formulas in each exercise by their numbered position in the list. Each student was expected to infer the syntax for a well-formed expression from the formulas

in the list.

At the time we introduced the Finding-Axioms Exercises, the students had learned a quantifier-free version of first-order logic, some elementary algebra and some Boolean algebra. Their teacher was an earlier version of the computer-assisted logic program in use at IMSSS since 1964 (Suppes and Binford, 1965; Suppes and Ihrke, 1970; Goldberg, 1971; Suppes, 1971). We could thus assume that the students were familiar with the operation of a teletypewriter (the student's input device) and with the manner in which proofs could be constructed on the computer. The students were adept at taking advantage of some of the more flexible features of the new instructional system, namely, with (a) naming and renaming formulas to find a suitable group to choose as the axioms, (b) making up lemmas to decrease the number of steps in some of the more complicated proofs, and (c) devising derivation problems with premises to test the effect of adding one or more axioms.

The four Finding-Axioms Exercises are shown in Figure 5. The students' previous experience with an axiomatization of the natural numbers made Exercise 1 conceptually easier and gave them an opportunity to adjust to the features of the new instructional system. Most of the college students completed all four exercises;

Exercise 4 was not given to the seventh graders because of the level of difficulty of the substitutions required.

Insert Figure 5 about here

In Tables 1-4 we show the axioms selected by each student for each exercise in the obvious matrix form. Variation in the selection of axioms occurred in all four exercises, probably least in the most trivial of the four, the first exercise. However, even here one, Student 11, managed to use only nine axioms. Tables 5-8 show the formulas used by each student to prove the theorems in each exercise.

Insert Tables 1-8 about here

In the case of Exercise 2, Students 12, 13 and 14, the seventh graders, did not finish, and the axioms they selected are not necessarily adequate--this is immediately obvious in the case of Student 13. These three young students did complete Exercise 3. The 11 college students, i.e., Students 1-11, all selected one of two sets of axioms; the choice was as evenly divided as possible with an odd number of students. Axioms 4 and 5 are definitional in character. By ordinary axiomatic standards the choice

Exercise 1. Natural Numbers

About 10 axioms seem to be needed. See what you can do.

1. X+1=Y+1 -> X=Y
2. NOT X+1=0
3. (NOT X=Y) -> (NOT X+1=Y+1)
4. X+0=X
5. NOT X+1=X
6. X+(Y+1)=(X+Y)+1
7. X+Y=Y+X
8. 0+X=X
9. Zx0 = 0
10. Z x(Y+1)=(Z x Y)+Z
11. Z x Y=Y x Z
12. 0 x Z=0
13. Z x 1=Z
14. NOT 0=1

Fig. 5. Finding-Axioms Exercises.

Exercise 2. Order Properties

Find 4 of the 8 statements that will serve as axioms. For intuitive purposes, think of P as greater than and Q as equal to or greater than.

1. X P Y & Y P Z -> X P Z

2. X P Y -> NOT Y P X

3. NOT X P X

4. X Q Y -> NOT Y P X

5. NOT X P Y -> Y Q X

6. X Q Y OR Y Q X

7. X Q Y & Y Q Z -> X Q Z

8. X Q X

Fig. 5, continued.

Exercise 3. Lattices

This exercise is close to the earlier work on Boolean algebra. The operations are the Boolean operations and the relation Q is like the relation of subset. But there is no complementation. We let ^ stand for intersection, and V for union.

1. X Q Y & Y Q Z -> X Q Z [Transitivity]
2. X Q Y & Y Q X -> X = Y [Antisymmetry]
3. X Q X [Reflexive]
4. X Q Y -> X ^ Y = X
5. X ^ Y = X -> X Q Y
6. X ^ X = X [Idempotent]
7. X V X = X
8. X ^ Y = Y ^ X [Commutative]
9. X V Y = Y V X
10. X ^ (Y ^ Z)=(X ^ Y)^ Z [Associative]
11. X V (Y V Z)=(X V Y)V Z
12. X ^ (X V Y)= X [Absorption]
13. X V (X ^ Y) = X
14. X Q Y & X Q Z -> X^Y = X^Z
15. X Q Y -> X V Y=Y
16. X V Y = Y -> X Q Y
17. X ^ Y Q X
18. X Q X V Y
19. X Q Z & Y Q Z -> X V Y Q Z

Fig. 5, continued.

20. X Q Y & X Q Z -> X Q Y ^ Z

21. X ^ Y Q X V Y

22. X ^ (X ^ Y) = X ^ Y

23. X ^ (Y ^ Z) Q Y

24. X ^ Y Q X V Z

25. X Q (Y ^ Z) V X

Exercise 4. Betweenness

B(X,Y,Z) means that Y is between X and Z on a line segment. We still call it betweenness when X=Y or Y=Z. Find 5 of the 11 statements as axioms.

1. B(X,X,X)

2. B(X,Y,X) -> X=Y

3. B(X,Y,Z) -> B(Z,Y,X)

4. X=Y -> B(X,Y,Z)

5. B(X,Y,W) & B(Y,Z,W) -> B(X,Y,Z)

6. (NOT Y=Z & B(X,Y,Z) & B(Y,Z,W)) -> B(X,Y,W)

7. B(X,Y,Z) & B(X,W,Z) -> B(Y,W,Z) OR B(W,Y,Z)

8. B(X,Y,Z) & B(Y,X,Z) -> X=Y

9. (B(X,Y,Z) OR B(Y,Z,X)) OR B(Z,X,Y)

10. ((NOT Y=Z) & B(X,Y,Z) & B(Y,Z,W)) -> B(X,Z,W)

11. B(X,Y,X) -> B(Z,Y,X)

Fig. 5, continued.

TABLE 1. EXERCISE 1

AXIOMS Student no. / Formula no.	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	X		X	X	X	X	X	X	X		X	X	X	X
2	X	X	X	X	X	X	X	X	X	X	X	X	X	X
3		X							X	X				
4	X	X	X	X		X	X	X	X	X	X	X	X	
5	X	X	X	X	X	X	X	X	X	X	X	X	X	X
6	X	X	X	X	X	X	X	X	X	X	X	X	X	X
7	X	X	X	X	X	X	X	X	X	X	X	X	X	X
8					X									X
9	X	X		X	X	X	X	X	X	X	X		X	X
10	X	X	X	X	X	X	X	X	X	X	X	X	X	X
11	X	X	X	X	X	X	X	X	X	X	X	X	X	X
12			X									X		
13	X	X		X	X	X	X	X		X		X	X	X
14			X											

TABLE 2. EXERCISE 2

AXIOMS Student no. / Formula no.	1	2	3	4	5	6	7	8	9	10	11	12*	13*	14*
1												X	X	
2	X		X	X			X					X		
3														X
4	X	X	X	X	X	X	X	X	X	X	X	X	X	
5	X	X	X	X	X	X	X	X	X	X	X	X		X
6		X			X	X		X	X	X	X		X	X
7	X	X	X	X	X	X	X	X	X	X	X			X
8													X	

TABLE 3. EXERCISE 3

AXIOMS / Student no. / Formula no.	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1									X					
2														
3														
4	X	X	X	X	X		X	X	X	X	X		X	X
5		X	X	X		X	X	X		X	X	X	X	X
6	X					X						X		
7														
8	X	X	X	X	X	X	X	X	X	X	X	X	X	X
9	X	X	X	X	X	X	X	X	X	X	X	X	X	X
10	X	X	X	X	X	X	X	X	X	X	X	X	X	X
11	X	X	X	X	X	X	X	X	X	X	X	X	X	X
12		X	X	X		X	X			X	X	X	X	X
13	X	X	X	X			X		X	X	X		X	X
14														
15					X	X		X				X		
16	X							X						
17					X									
18					X									
.														
.														
.														
25									X					

TABLE 4. EXERCISE 4

AXIOMS / Student no. / Formula no.	1	2	3	4	5	6	7	8	9	10	11
1											
2			X	X		X			X	X	X
3		X	X	X		X		X	X	X	X
4								X			
5		X	X	X		X		X	X	X	X
6		X								X	
7											
8		X						X			
9		X	X	X		X		X	X	X	X
10			X	X		X			X		X
11											

TABLE 5. EXERCISE 1

THEOREMS

Student no.	Formula no.	Proved using formulas no.	Student no.	Formula no.	Proved using formulas no.
1	3	1	8	14	4,5
	8	4,7		8	4,7
	12	9,11		12	9,11
	14	5,12		3	1
2	1	3	9	8	4,7
	8	4,7		14	5,8
	12	9,11		13	8,9,10
	14	4,5		12	9,11
3	13	4,7,10,11,12	10	1	3
	8	4,7		8	4,7
	9	11,12		12	9,11
	3	1		14	4,5
4	8	4,7	11**	8	4,7
	12	9,11		12	9,11
	14	4,5,7		13	8,9,10
	3	1		3	10
				14	2,8
5	4	7,8	12	9	11,12
	12	9,11		8	4,7
	3	1		3	1
	14	4,5		14	2,4,5
6	3	1	13	8	4,7
	8	4,7		12	9,11
	14	5,8		14	4,5
	12	9,11		3	1
7	8	4,7	14	4	7,8
	3	1		12	9,11
	12	9,11		3	4,7,8,9,10
	14	5,7,8		14	4,5

TABLE 6. EXERCISE 2

THEOREMS

Student no.	Formula no.	Proved using formulas no.	Student no.	Formula no.	Proved using formulas no.
1	1	2,4,5,7	8	8	6
	3	2		3	4,8
	6	2,5		2	L1,6
	8	4,5		1	2,4,5,6,7
2	8	6	9	2	L1,L4,4
	3	4,6		3	2
	2	4,6		8	3,5
	1	2,4,5,7		1	L3,4,5,7
3	6	2,5	10	1	2,4,5,7
	3	2		3	2
	8	6		6	2,5
	1	2,4,5,7		8	3,5
4	3	2	11	8	6
	8	3,5		3	4,8
	6	2,5		2	L1,4,6
	1	2,4,5,7		1	L3,4,5,7
5	8	6	12	3	2
	3	4		8	5,3
	1	4,5,6,7		6*	
	2	1,3		7*	
6	2	4,6	13	3	4,8
	1	2,4,5		2	1,3
	3	2		5*	
	8	3,5		7*	
7	3	2	14	8	3,5
	8	3,5		1*	
	6	3,5		2	1,3
	1	2,4,5,7		4*	

TABLE 7. EXERCISE 3

THEOREMS

Student no.	Formula no.	Proved using formula no.	Student no.	Formula no.	Proved using formula no.
1	15	4,8,9,13	8	2	9,15
	5	8,9,13,16		19	L1,11,15
	1	4,5,10		20	L2,4,10
	2	9,15		3	L3
	3	5,6		7	3,15
	7	3,15		6	3,4
	18	7,11,16		22	6,10
	12	4,18		17	5,8,22
	14	4		18	7,11,16
	22	6,10		12	4,18
	17	5,8,22		13	9,15,17
	19	11,15,16		21	9,11,13,16
	20	4,5,8,10		14	4
	21	9,11,13,16		23	5,6,8,10
	23	5,6,8,10		24	5,8,10,12
	24	9,11,13,16		25	5,9,12
	25	5,9,12		1	4,5,10
2	1	4,5,10	9	2	4,8
	2	4,8		3	9,13,25
	6	12,13		6	3,4
	3	5,6		7	6,13
	16	5,12		5	6,8,13,25
	15	4,9,8,13		14	4
	7	3,15		15	4,8,9,13
	14	4		16	6,9,25
	17	13,16		17	6,13,25
	18	5,12		18	6,9,25
	19	11,15,16		19	11,15,18
	20	4,5,8,10		20	4,5,8,10
	21	1,17,18		12	4,6,9,25
	22	6,10		21	5,9,10,12
	23	8,10,17		22	4,8,17
	24	1,17,18		23	8,10,17
	25	9,18		24	10,12,23

TABLE 7, continued.

3	15	4,8,9,13	10	1	4,5,10
	16	5,12		2	4,8
	6	12,13		6	8,12,13
	7	12,13		7	3,15
	2	4,8		14	4
	14	4		17	5,6,8,10
	24	9,11,13,16		18	7,11,16
	17	7,24		19	11,15,16
	18	6,24		20	4,5,8,10
	25	6,9,24		21	1,17,18
	23	24,10,8,7		22	10,6
	22	6,10		23	8,10,17
	21	24		24	1,17,18
	1	5,11,12,15		25	9,18
	3	7,16		15	4,8,9,13
	19	11,15,16		16	5,12
	20	4,5,10		3	5,6
4	1	4,5,10	11	6	12,13
	14	4		7	12,13
	2	4,8		3	5,6
	20	4,5,8,10		2	4,8
	19	11,15,16		14	4
	15	4,8,9,13		22	6,10
	6	12,13		23	5,6,8,10
	3	5,6		17	5,6,8,10
	7	3,15		18	5,12
	18	7,11,16		15	4,9,13
	17	5,6,8,10		16	5,12
	21	5,9,10,12		25	7,9,11,16
	22	6,10		24	5,8,10,12
	23	5,6,8,10		21	5,9,10,12
	24	11,13,16		19	11,15,16
	25	7,9,11,16		20	4,5,10
	16	5,12		1	4,5,10

TABLE 7, continued.

5	2	4
	14	4
	25	9,18
	5	8,17
	16	18
	12	4,18
	13	9,15,17
	23	8,10,17
	20	4,5,10
	19	11,12,16
	1	4,8,23
	3	13,25
	4	12,15
	3	5,6
	6	3,4
	22	6,10
	7	3,12
6	16	5,12
	4	12,15
	3	5,6
	7	10,15
	1	11,15,16
	2	9,15
	17	5,6,8,10
	13	9,15,17
	14	4
	18	5,12
	19	11,15,16
	20	4,5,10
	24	9,11,13,16
	21	24
	22	4,8,17
	23	8,10,17
	25	9,18
12	22	6,10
	14	4
	16	5,12
	3	5,6
	2	3,4,6,8,14
	1	4,5,10
	18	7,11,16
	19	10,11,15,16
	7	3,15
	20	4,5,10
	17	5,6,8,10
	21	1,17,18
	23	8,10,17
	25	9,18
	4	5,12
	13	9,15,17
	24	1,17,18
13	6	3,4
	2	4,6,8,10
	7	3,15
	1	4,5,10
	14	4,8,10
	16	5,12
	20	4,5,10
	17	9,12,13,16
	18	5,12
	19	4,11,15,16
	22	4,8,17
	3	5,12,13
	15	4,8,9,13
	21	1,16,17,18
	23	8,10,17
	24*	
	25*	

TABLE 7, continued.

7	3	5,12,13	14	7	3,15
	6	3,4		1	4,5,10
	25	5,9,12		14	4
	1	4,5,10		2	4,8,10
	24	8,10,12		16	5,12
	14	4		17	3,9,13,16
	2	4,8		18	5,12
	18	5,12		15	4,8,9,13,17
	15	4,8,9,13		6	7,12
	16	5,12		3	5,12,13
	7	15		19	4,9,10,11, 12,15,16
	17	9,11,16		20	4,5,10
	19	11,15,16		22	6,10
	20	4,5,10		21	1,17,18
	21	1,17,18		23	8,10,17
	22	6,10		24*	
	23	8,10,17		25*	

TABLE 8. EXERCISE 4

THEOREMS

Student no.	Formula no.	Proved using formulas no.	Student no.	Formula no.	Proved using formulas no.
2	1	9	8	1	4
	2	1,3,5,8		2	3,4,8
	4	2,3,9		11	2,3,4
	10	3,6		6	L5,L4,3,8,9
	11	2,3,4		10	L1,L2,3,8,9
	7	2,3,4,6,9,10		7*	
3	1	9	9	1	9
	8	2,5		8	2,5
	4	1,2,3,9		11	2,3,9
	11	2,3,4		4	1,3,11
	6	3,10		7	L1,3,8,9,10
	7	L2,2,3,9,10		6	L1,3,5,8,9,10
4	4	2,3,9	10	1	9
	1	4		4	L1
	11	2,3,4		8	2,5
	8	2,5		11	2,3,4
	6	3,10		7	L3,L4,L5,3,5,9
	7	2,3,4,5,9,10		10	L1,2,3,5,6,9
6	1	9	11	4	2,3,9
	11	L1,2,3		1	4
	4	1,3,11		11	2,3,4
	6	3,10		8	2,5
	8	2,5		6	3,10
	7	L1,3,8,9,10		7*	

of Formulas 2 and 7 as the remaining axioms is somewhat surprising; certainly they are not a choice that would be found in any text dealing with these elementary order properties.

Exercise 3 on lattices produced the greatest variety of axioms. Seven students of the 14 used the same set, but the remaining seven students produced five additional sets of axioms. Student 8 introduced the following three lemmas to shorten his proofs.

Lemma 1: (NOT X Q Y)-> (NOT X V Y = Y)

Lemma 2: (NOT X Q Y)-> (NOT X ^ Y = X)

Lemma 3: (X=Y)-> (X Q Y) & (Y Q X)

Because of its greater difficulty, Exercise 4 was completed by only six of the 11 college students. (The college course required completion of this exercise only by those students who wanted the highest grade.) Students 8 and 11 selected adequate axioms, but did not complete the proofs. Various students introduced and used the following five lemmas.

Lemma 1: B(X,X,Y)

Lemma 2: B(X,Y,Y)

Lemma 3: NOT (B(Y,W,Z) OR B(W,Y,Z))->
NOT B(Y,W,Z)

Lemma 4: NOT B(Z,Y,X)-> NOT B(X,Y,Z)

Lemma 5: NOT B(Z,Y,W)-> NOT Z=Y

We have looked at the data on the order in which

formulas that were not selected as axioms were proved by each student. We also have looked at the sequence of rejection of formulas as possible axioms, but the data are too elaborate to reproduce here, and thus we have restricted ourselves to points that seem of particular interest.

In conversations with the students we found that they experienced some difficulty in crossing over from one exercise to another when identical symbols were used (namely, Q). There was also an initial desire on the part of the seventh graders to use theorems from one exercise for proving theorems in another. This confusion in distinguishing between the distinct theories might stem from the CAI environment in which they had been studying earlier, one in which the curriculum was designed so that each axiom introduced and each theorem proved were always available for use in subsequent proofs.

To find a suitable set of axioms for a given exercise, the students had to experiment with different combinations of the formulas, perhaps trying to construct proofs for formulas already designated as axioms. In doing this, they sometimes made an error that we term "cycling"--proving Formula A from B designated as an axiom, and then proving B from the already established Theorem A. A trial-and-error heuristic for choosing axioms was developed

by the students from this cycling problem: if the proofs for two formulas depend on one another, assume that one should be an axiom. We offered another heuristic: to be able to prove things about the relationships between the variables, pick as axioms those formulas that define the relations in terms of one another, not just singularly.

The longest proofs were of formulas (7) and (10) of Exercise 4. Each of the 11 college students attempted to find a proof of Formula (7). The mean length of these proofs was 45 lines, with the shortest, which used three lemmas, being 39 lines, and the longest being 77 lines without using any lemmas. Only three students proved Formula (10), with proofs of 17, 32 and 43 lines. The rather elegant shortest proof of 17 lines is shown as Figure 6.

Insert Figure 6 about here

The general response of the college students to these exercises was surprisingly positive. Even though the Finding-Axioms Exercises were considerably harder than the earlier exercises in the course, all of the students expressed a clear preference for the Finding-Axioms Exercises, because of the apparent greater interest in the exploratory search for axioms as opposed to straight

```
:PROVE: 10
PROVE      (((NOT Y=Z)& B(X,Y,Z))& B(Y,Z,W))-> B(X,Z,W)

:WP        (1)        ((NOT Y=Z)& B(X,Y,Z))& B(Y,Z,W)

:1RC       (2)        B(Y,Z,W)

:AXB     B(X,Y,Z)-> B(Z,Y,X)
X::Y
Y::Z
Z::W       (3)        B(Y,Z,W)-> B(W,Z,Y)

:3,2AA     (4)        B(W,Z,Y)

:1LC       (5)        (NOT Y=Z)& B(X,Y,Z)

:5RC       (6)        B(X,Y,Z)

:AXB     B(X,Y,Z)-> B(Z,Y,X)
X::X
Y::Y
Z::Z       (7)        B(X,Y,Z)-> B(Z,Y,X)

:7,6AA     (8)        B(Z,Y,X)

:5LC       (9)        NOT Y=Z

:9CE1      (10)       NOT Z=Y

:AXB       (((NOT Y=Z)& B(X,Y,Z))& B(Y,Z,W))-> B(X,Y,W)
W::X
X::W
Y::Z
Z::Y       (11)       (((NOT Z=Y)& B(W,Z,Y))& B(Z,Y,X))-> B(W,Z,X)

:10,4FC    (12)       (NOT Z=Y)& B(W,Z,Y)

:12,8FC    (13)       ((NOT Z=Y)& B(W,Z,Y))& B(Z,Y,X)
```

Fig. 6. Shortest student proof for formula (10), Exercise 4.
AXA: (((NOT X=Y)& B(Z,X,Y))& B(X,Y,W))->B(Z,X,W)
AXB: B(X,Y,Z)-> B(Z,Y,X).

```
:11.13AA  (14)      B(W,Z,X)

:AXB     B(X,Y,Z)-> B(Z,Y,X)
X::W
Y::Z
Z::X      (15)      B(W,Z,X)-> B(X,Z,W)

:15.14AA  (16)      B(X,Z,W)

:1.16CP   (17)  (((NOT Y=Z)& B(X,Y,Z))& B(Y,Z,W))-> B(X,Z,W)
CORRECT...
```

Fig. 6, continued.

derivations from given axioms or premises.

The reaction of the 12-year-old students was rather different. They had had extensive training in proving theorems from given axioms, and they felt uneasy at not knowing immediately where to begin, or if once they started, whether their initial subset of axioms was actually adequate for what they wanted to prove. The currect revision of the course consequently includes counterexamples and very elementary Finding-Axioms Exercises almost from the beginning.

3. Theorem Provers for Instructional Use

The reader might question the instructional worth of having a student prove complex theorems within the framework of a proof checker which, as so far illustrated, expects the user to construct rigorous proofs. As discussed in Section 2, the limitation of the program for teaching mathematics is just this requirement that the student construct an explicit formal proof for every theorem. The routine steps of more advanced mathematical work must be compressed and eliminated from the student's explicit focus of concern in order to provide adequate time to concentrate on the crucial conceptual steps in a given proof.

A significant contribution of the system to the development of more advanced mathematics courses in a CAI

environment is the use of theorem provers for instructional purposes. With theorem provers, the student can instruct the program to move from one point to another in the proof without explicitly carrying out the mediating steps. The intention is that the skipped steps be modest and of the right level of difficulty for mechanical theorem provers of a noninteractive nature. An example is repeated use of the commutative and associative laws in a fashion that is common in elementary algebraic arguments.

Using the instructional system as a research tool, we have been able to study possible roles mechanical theorem provers can play in the construction of mathematically valid proofs by the student. As modes of operation for the use of theorem provers, we introduced generalized interchange laws and instantiation rules for each axiom and proved theorem, as well as an instruction called SHOW. For example, whenever a new expression in the form of an identity or biconditional is established as a true statement of a given theory, and a name is assigned to it, the program is capable of performing both the substitution and replacement rules in one step.

The automatic generalization rule for formulas in the form of a conditional represents a definite savings in the number of steps required to complete a proof, as illustrated in Figure 7 by the derivation of line (7) from

(4) and Theorem A, line (8) from (5) and Theorem A, and line (9) from (6), (7), (8), and Axiom TRA. The proof depends directly on one previous theorem and one axiom, which are shown at the bottom of the figure and which are part of a system of constructive plane geometry. The user simply references the name of the formula and an ordered list of proof lines whose conjunction is a substitution instance of the antecedent of the formula. The program can then generate the corresponding instance of the consequent. If substitutable variables occurring in the consequent do not occur in the antecedent, the program will ask the student to complete the desired substitution. The ability to derive new inference rules extends this instantiation rule in enabling the student to detach from a theorem a formula that would otherwise only be obtainable by iterative application of modus ponens on an instance of that theorem.

Insert Figure 7 about here

Our efforts to interface the instructional system with the theorem-proving program of Allen and Luckham (1970) have been moderately successful. Basically, our idea is to let the student type a line into the proof, thereby claiming that it is a valid inference from the work

```
PROVE      (((NOT Y=Z)& B(X,Y,Z))& B(Y,Z,W))-> B(X,Z,W)
:WP        (1)        ((NOT Y=Z)& B(X,Y,Z))& B(Y,Z,W)
:1LC       (2)        (NOT Y=Z)& B(X,Y,Z)
:2LC       (3)        NOT Y=Z
:2RC       (4)        B(X,Y,Z)
:1RC       (5)        B(Y,Z,W)
:3CE1      (6)        NOT Z=Y
:4THA      (7)        B(Z,Y,X)
:5THA      (8)        B(W,Z,Y)
:6.8.7TRA  (9)        B(W,Z,X)
:9THA      (10)       B(X,Z,W)
:1.10CP    (11)  (((NOT Y=Z)& B(X,Y,Z))& B(Y,Z,W))-> B(X,Z,W)
CORRECT...
```

Fig. 7. A proof using general substitution rules.
Theorem A: B(X,Y,Z)->(B(Z,Y,X)
Axiom TRA: (((NOT X=Y)& B(W,X,Y))& (B(X,Y,Z))->B(W,X,Z)

he has already done. He then calls on the theorem prover to verify his claim. The student also must indicate which lines already in the proof and what instances of axioms and theorems he thinks should enter into the theorem prover's computations. Formula (10) from the fourth Finding-Axioms Exercise is presented as an example of using the SHOW rule (Figure 8). The SHOW rule, together with those discussed above, were used to eliminate ten lines from the minimum student proof shown in Figure 6.

Insert Figure 8 about here

A second and closely related activity in which theorem provers are useful is that of monitoring the student's activity while he is in the process of searching for a proof and then giving him hints of how he may complete the proof he has begun. At least in elementary domains of mathematics this role of a theorem prover has already been implemented as an instruction called HELP. A heuristically based theorem prover was designed to perform the work the student is expected to do, i.e., it constructs proofs in the elementary theory of Abelian groups. By taking the steps of the student's partial or erroneous work into account when searching for a solution, the theorem prover can compute various ways to complete the student's

```
PROVE      (((NOT Y=Z)& B(X,Y,Z))& B(Y,Z,W))-> B(X,Z,W)

:WP        (1)        ((NOT Y=Z)& B(X,Y,Z))& B(Y,Z,W)

:1RC       (2)        B(Y,Z,W)

:2AXB      (3)        B(W,Z,Y)

:SHOW      (4)        ((NOT Z=Y)& B(W,Z,Y))& B(Z,Y,X)
FROM LINES OF THE DERIVATION?
::1,3
FROM AXIOMS OR THEOREMS?
::AXB    B(X,Y,Z)-> B(Z,Y,X)
X::X
Y::Y
Z::Z
::
OK?  Y
LINE 4 IS OK

:4AXA      (5)        B(W,Z,X)

:5AXB      (6)        B(X,Z,W)

:1,6CP     (7)   (((NOT Y=Z)& B(X,Y,Z))& B(Y,Z,W))-> B(X,Z,W)

CORRECT...
```

Fig. 8. Proof of Formula (10), Exercise 4, using the SHOW instruction.
AXA: (((NOT X=Y)& B(Z,X,Y))& B(X,Y,W))-> B(Z,X,W)
AXB: B(X,Y,Z)-> B(Z,Y,X)

task. From this information, the instructional system can generate a tutorial dialogue aimed at helping the student construct a successful proof. The details of using a theorem prover as a proof analyzer to help a student continue his work is dealt with elsewhere (Goldberg, forthcoming).

List of References

Allen, J., & Luckham, D. An interactive theorem-proving program. In B. Meltzer & D. Michie (Eds.), Machine Intelligence 5. New York: American Elsevier, 1970.

Goldberg, A. A generalized instructional system for teaching elementary mathematical logic. Technical Report No. 179, October 11, 1971, Stanford University, Institute for Mathematical Studies in the Social Sciences.

Goldberg, A. Computer-assisted instruction: The application of theorem-proving to adaptive response analysis. Doctoral dissertation, University of Chicago, Chicago, Illinois, forthcoming.

Suppes, P. Computer-assisted instruction at Stanford. Technical Report No. 174, May 19, 1971, Stanford University, Institute for Mathematical Studies in the Social Sciences.

Suppes, P., & Binford, F. Experimental teaching of mathematical logic in the elementary school. The Arithmetic Teacher, 1965, 12, 187-195.

Suppes, P., & Ihrke, C. Accelerated program in elementary-school mathematics--the fourth year. Psychology in the Schools, 1970, 7, 111-126.

(Continued from inside front cover)

96 R. C. Atkinson, J. W. Brelsford, and R. M. Shiffrin. Multi-process models for memory with applications to a continuous presentation task. April 13, 1966. (J. math. Psychol., 1967, 4, 277-300).
97 P. Suppes and E. Crothers. Some remarks on stimulus-response theories of language learning. June 12, 1966.
98 R. Bjork. All-or-none subprocesses in the learning of complex sequences. (J. math. Psychol., 1968, 1, 182-195).
99 E. Gammon. The statistical determination of linguistic units. July 1, 1966.
100 P. Suppes, L. Hyman, and M. Jerman. Linear structural models for response and latency performance in arithmetic. (In J. P. Hill (ed.), Minnesota Symposia on Child Psychology. Minneapolis, Minn.: 1967. Pp. 160-200).
101 J. L. Young. Effects of intervals between reinforcements and test trials in paired-associate learning. August 1, 1966.
102 H. A. Wilson. An investigation of linguistic unit size in memory processes. August 3, 1966.
103 J. T. Townsend. Choice behavior in a cued-recognition task. August 8, 1966.
104 W. H. Batchelder. A mathematical analysis of multi-level verbal learning. August 9, 1966.
105 H. A. Taylor. The observing response in a cued psychophysical task. August 10, 1966.
106 R. A. Bjork. Learning and short-term retention of paired associates in relation to specific sequences of interpresentation intervals. August 11, 1966.
107 R. C. Atkinson and R. M. Shiffrin. Some Two-process models for memory. September 30, 1966.
108 P. Suppes and C. Ihrke. Accelerated program in elementary-school mathematics--the third year. January 30, 1967.
109 P. Suppes and I. Rosenthal-Hill. Concept formation by kindergarten children in a card-sorting task. February 27, 1967.
110 R. C. Atkinson and R. M. Shiffrin. Human memory: a proposed system and its control processes. March 21, 1967.
111 Theodore S. Rodgers. Linguistic considerations in the design of the Stanford computer-based curriculum in initial reading. June 1, 1967.
112 Jack M. Knutsen. Spelling drills using a computer-assisted instructional system. June 30, 1967.
113 R. C. Atkinson. Instruction in initial reading under computer control: the Stanford Project. July 14, 1967.
114 J. W. Brelsford, Jr. and R. C. Atkinson. Recall of paired-associates as a function of overt and covert rehearsal procedures. July 21, 1967.
115 J. H. Stalzer. Some results concerning subjective probability structures with semiorders. August 1, 1967
116 D. E. Rumelhart. The effects of interpresentation intervals on performance in a continuous paired-associate task. August 11, 1967.
117 E. J. Fishman, L. Keller, and R. E. Atkinson. Massed vs. distributed practice in computerized spelling drills. August 18, 1967.
118 G. J. Groen. An investigation of some counting algorithms for simple addition problems. August 21, 1967.
119 H. A. Wilson and R. C. Atkinson. Computer-based instruction in initial reading: a progress report on the Stanford Project. August 25, 1967.
120 F. S. Roberts and P. Suppes. Some problems in the geometry of visual perception. August 31, 1967. (Synthese, 1967, 17, 173-201)
121 D. Jamison. Bayesian decisions under total and partial ignorance. D. Jamison and J. Kozielecki. Subjective probabilities under total uncertainty. September 4, 1967.
122 R. C. Atkinson. Computerized instruction and the learning process. September 15, 1967.
123 W. K. Estes. Outline of a theory of punishment. October 1, 1967.
124 T. S. Rodgers. Measuring vocabulary difficulty: An analysis of item variables in learning Russian-English and Japanese-English vocabulary parts. December 18, 1967.
125 W. K. Estes. Reinforcement in human learning. December 20, 1967.
126 G. L. Wolford, D. L. Wessel, W. K. Estes. Further evidence concerning scanning and sampling assumptions of visual detection models. January 31, 1968.
127 R. C. Atkinson and R. M. Shiffrin. Some speculations on storage and retrieval processes in long-term memory. February 2, 1968.
128 John Holmgren. Visual detection with imperfect recognition. March 29, 1968.
129 Lucille B. Mlodnosky. The Frostig and the Bender Gestalt as predictors of reading achievement. April 12, 1968.
130 P. Suppes. Some theoretical models for mathematics learning. April 15, 1968. (Journal of Research and Development in Education, 1967, 1, 5-22)
131 G. M. Olson. Learning and retention in a continuous recognition task. May 15, 1968.
132 Ruth Norene Hartley. An investigation of list types and cues to facilitate initial reading vocabulary acquisition. May 29, 1968.
133 P. Suppes. Stimulus-response theory of finite automata. June 19, 1968.
134 N. Moler and P. Suppes. Quantifier-free axioms for constructive plane geometry. June 20, 1968. (In J. C. H. Gerretsen and F. Oort (Eds.), Compositio Mathematica. Vol. 20. Groningen, The Netherlands: Wolters-Noordhoff, 1968. Pp. 143-152.)
135 W. K. Estes and D. P. Horst. Latency as a function of number of response alternatives in paired-associate learning. July 1, 1968.
136 M. Schlag-Rey and P. Suppes. High-order dimensions in concept identification. July 2, 1968. (Psychon. Sci., 1968, 11, 141-142)
137 R. M. Shiffrin. Search and retrieval processes in long-term memory. August 15, 1968.
138 R. D. Freund, G. R. Loftus, and R.C. Atkinson. Applications of multiprocess models for memory to continuous recognition tasks. December 18, 1968.
139 R. C. Atkinson. Information delay in human learning. December 18, 1968.
140* R. C. Atkinson, J. E. Holmgren, and J. F. Juola. Processing time as influenced by the number of elements in the visual display. March 14, 1969.
141 P. Suppes, E. F. Loftus, and M. Jerman. Problem-solving on a computer-based teletype. March 25, 1969.
142 P. Suppes and Mona Morningstar. Evaluation of three computer-assisted instruction programs. May 2, 1969.
143 P. Suppes. On the problems of using mathematics in the development of the social sciences. May 12, 1969.
144 Z. Domotor. Probabilistic relational structures and their applications. May 14, 1969.
145 R. C. Atkinson and T. D. Wickens. Human memory and the concept of reinforcement. May 20, 1969.
146 R. J. Titiev. Some model-theoretic results in measurement theory. May 22, 1969.
147 P. Suppes. Measurement: Problems of theory and application. June 12, 1969.
148 P. Suppes and C. Ihrke. Accelerated program in elementary-school mathematics--the fourth year. August 7, 1969.
149 D. Rundus and R.C. Atkinson. Rehearsal in free recall: A procedure for direct observation. August 12, 1969.
150 P. Suppes and S. Feldman. Young children's comprehension of logical connectives. October 15, 1969.

(Continued on back cover)

(Continued from inside back cover)

151 Joaquim H. Laubsch. An adaptive teaching system for optimal item allocation. November 14, 1969.
152 Roberta L. Klatzky and Richard C. Atkinson. Memory scans based on alternative test stimulus representations. November 25, 1969.
153 John E. Holmgren. Response latency as an indicant of information processing in visual search tasks. March 16, 1970.
154 Patrick Suppes. Probabilistic grammars for natural languages. May 15, 1970.
155 E. Gammon. A syntactical analysis of some first-grade readers. June 22, 1970.
156 Kenneth N. Wexler. An automaton analysis of the learning of a nimiature system of Japanese. July 24, 1970.
157 R. C. Atkinson and J.A. Paulson. An approach to the psychology of instruction. August 14, 1970.
158 R.C. Atkinson, J.D. Fletcher, H.C. Chetin, and C.M. Stauffer. Instruction in initial reading under computer control: the Stanford project. August 13, 1970.
159 Dewey J. Rundus. An analysis of rehearsal processes in free recall. August 21, 1970.
160 R.L. Klatzky, J.F. Juola, and R.C. Atkinson. Test stimulus representation and experimental context effects in memory scanning.
161 William A. Rottmayer. A formal theory of perception. November 13, 1970.
162 Elizabeth Jane Fishman Loftus. An analysis of the structural variables that determine problem-solving difficulty on a computer-based teletype. December 18, 1970.
163 Joseph A. Van Campen. Towards the automatic generation of programmed foreign-language instructional materials. January 11, 1971.
164 Jamesine Friend and R.C. Atkinson. Computer-assisted instruction in programming: AID. January 25, 1971.
165 Lawrence James Hubert. A formal model for the perceptual processing of geometric configurations. February 19, 1971.
166 J. F. Juola, I.S. Fischler, C.T. Wood, and R.C. Atkinson. Recognition time for information stored in long-term memory.
167 R.L. Klatzky and R.C. Atkinson. Specialization of the cerebral hemispheres in scanning for information in short-term memory.
168 J.D. Fletcher and R.C. Atkinson. An evaluation of the Stanford CAI program in initial reading (grades K through 3). March 12, 1971.
169 James F. Juola and R.C. Atkinson. Memory scanning for words versus categories.
170 Ira S. Fischler and James F. Juola. Effects of repeated tests on recognition time for information in long-term memory.
171 Patrick Suppes. Semantics of context-free fragments of natural languages. March 30, 1971.
172 Jamesine Friend. Instruct coders' manual. May 1, 1971.
173 R.C. Atkinson and R.M. Shiffrin. The control processes of short-term memory. April 19, 1971.
174 Patrick Suppes. Computer-assisted instruction at Stanford. May 19, 1971.
175 D. Jamison, J.D. Fletcher, P. Suppes and R.C.Atkinson. Cost and performance of computer-assisted instruction for compensatory education.
176 Joseph Offir. Some mathematical models of individual differences in learning and performance. June 28, 1971.
177 Richard C. Atkinson and James F. Juola. Factors influencing speed and accuracy of word recognition. August 12, 1971.
178 P. Suppes, A. Goldberg, G. Kanz, B. Searle and C. Stauffer. Teacher's handbook for CAI courses. September 1, 1971.
179 Adele Goldberg. A generalized instructional system for elementary mathematical logic. October 11, 1971.
180 Max Jerman. Instruction in problem solving and an analysis of structural variables that contribute to problem-solving difficulty. November 12, 1971.
181 Patrick Suppes. On the grammar and model-theoretic semantics of children's noun phrases. November 29, 1971.
182 Georg Kreisel. Five notes on the application of proof theory to computer science. December 10, 1971.
183 James Michael Moloney. An investigation of college student performance on a logic curriculum in a computer-assisted instruction setting. January 28, 1972.
184 J.E. Friend, J.D. Fletcher and R.C. Atkinson. Student performance in computer-assisted instruction in programming. May 10, 1972.
185 Robert Lawrence Smith, Jr. The syntax and semantics of ERICA. June 14, 1972.
186 Adele Goldberg and Patrick Suppes. A computer-assisted instruction program for exercises on finding axioms. June 23, 1972.